

RANCANG BANGUN PROTOTIPE SISTEM PENGATURAN PENYALAN LAMPU MENGGUNAKAN TWITTER

Arief Hendra Saptadi

Program Stud D-III Teknik Telekomunikasi Sekolah Tinggi Teknologi Telematika TELKOM
Jl. D. I. Panjaitan No. 128 Purwokerto 53147
ariefhs@stttelematikatelkom.ac.id

ABSTRAK

Dewasa ini karena berbagai kesibukan, tidak jarang rumah ditinggalkan oleh penghuninya. Untuk mengantisipasi kegelapan di waktu malam, penghuni biasanya cenderung untuk menyalakan lampu luar di siang hari terlebih dahulu sebelum meninggalkan rumah. Selain tidak hemat energi, cara seperti ini juga dapat memberikan petunjuk kepada pihak-pihak yang tidak bertanggung jawab bahwa rumah dalam kondisi kosong. Tujuan dari penelitian ini adalah melakukan rancang bangun sebuah prototipe sistem pengaturan penyalan lampu menggunakan Twitter. Twitter sebagai salah satu aplikasi media sosial menyediakan API (*Application Programming Interface*) yang dapat diakses oleh aplikasi server. Aplikasi di PC ini selanjutnya memberikan sinyal perintah kepada sistem mikropengendali untuk menyalakan atau mematikan lampu sesuai isi teks status (tweet). Berdasarkan pengujian diperoleh hasil bahwa query yang dikirimkan ke Twitter memiliki tingkat kesuksesan 100% dengan waktu respon rata-rata 0,286 detik. Tiga buah lampu yang dihubungkan ke mikro pengendali ATmega328P telah dapat dikendalikan sesuai isi tweet. Kendatipun demikian jika penghuni meninggalkan rumah untuk waktu lama, maka harus menyalakan dan mematikan lampu secara berulang-ulang sehingga ke depannya perlu dikembangkan sistem penjadwalan yang dapat diatur melalui tweet.

Kata kunci: prototipe sistem, penyalan lampu, Twitter

ABSTRACT

*Recently, due to various activities, the home is not rarely left for work by the occupant. To anticipate darkness at night, the occupant usually tends to switch the outdoor lamps on before leaving the home. Despite of being not energy-saving, this would bring a clue to irresponsible persons that the home is unoccupied. The purpose of this research is to design and build a prototype for lamp switching configuration system by using Twitter. Twitter as one of social media applications provides API (*Application Programming Interface*) which is accessible from a server application. Subsequently, this PC application transmits a command signal to a microcontroller system to switch the lamp on or off according to the content of status text (tweet). Based on performed tests, queries sent to Twitter had a success rate of 100% and average response time of 0.286 second. Three lamps connected to an ATmega328P microcontroller had been able to be controlled based on a tweet content. Nevertheless, if an occupant leaves the home for a long period of time, the lamps need to be switched on and off regularly, in which a scheduling system configurable through tweets needs to be developed.*

Keywords : system prototype, lamp switching, Twitter

PENDAHULUAN

Seiring dengan tuntutan pekerjaan, tidak jarang rumah ditinggalkan dalam keadaan kosong oleh penghuninya untuk bekerja. Sebagai upaya untuk memberikan penerangan di malam hari sementara si pemilik rumah harus bepergian di siang harinya dan pulang

larut malam atau keesokan paginya, maka tidak jarang lampu rumah dinyalakan dahulu. Kondisi lampu menyala di siang hari selain memunculkan isu tidak hemat energi, di sisi lain juga menimbulkan potensi kerawanan. Bagi pihak-pihak yang tidak bertanggung jawab, kondisi tersebut memberikan petunjuk

bahwa rumah dalam keadaan kosong, tak berpenghuni.

Salah satu solusi untuk permasalahan ini adalah dengan membuat sistem penyalan lampu yang dikendalikan dari jarak jauh. Media komunikasi yang ideal antara pengguna dengan sistem yang dijalankan adalah melalui internet. Hal ini seiring dengan pertumbuhan pengguna internet di Indonesia yang pada tahun 2015 ini diperkirakan sejumlah 93,4 juta orang, naik 11,5% dari tahun sebelumnya yaitu 83,7 juta orang (Statista, 2015). Dari seluruh pengguna yang mengakses web, 60% diantaranya menggunakan perangkat bergerak (*mobile*). Ada pun 21% dari keseluruhan aktivitas yang dilakukan oleh pengguna perangkat bergerak tadi adalah untuk mengakses media sosial (Inmobi dan Decisionfuel, 2014). Berdasarkan data tersebut maka dapat disimpulkan bahwa solusi pengendalian perangkat jarak jauh yang melibatkan perangkat bergerak dengan akses internet menuju media sosial adalah layak untuk diterapkan.

Twitter dan Internet of Things

Twitter sebagai salah satu media sosial terkemuka, menyediakan API (*Application Programming Interface*) yang dapat diakses oleh pengembang. Berbekal Twitter API tersebut, berbagai aplikasi dapat dikembangkan, termasuk di antaranya adalah aplikasi yang menjadi bagian dari sistem pengendalian jarak jauh.

Twitter sendiri sangat mendukung untuk dijadikan bagian dari suatu sistem waktu-nyata (*real time*) yang melibatkan berbagai perangkat sensor. Hal ini karena karakteristiknya sebagai platform *micro blogging* – sepanjang 140 karakter untuk setiap pesan – yang memungkinkan pelaporan (*report*) suatu *event*. Di samping itu, pesan yang ditampilkan di Twitter juga dapat diakses oleh beragam aplikasi klien di PC maupun di ponsel cerdas dan terdapat mekanisme pembatasan akses melalui relasi *following* dan *followers* (Yonezawa dan Tokuda, 2010).

Keberadaan layanan web yang berinteraksi dengan berbagai perangkat elektronik dan objek-objek fisik melalui medium internet ini mewujudkan apa yang disebut dengan *Internet of Things* (IoT). Situs jejaring sosial tidak hanya menjadi sarana berkomunikasi antar penggunanya namun juga

akan bertransformasi menjadi bagian (*thing*) dari objek-objek cerdas (*smart objects*) yang saling terhubung (Gubbi, dkk, 2013).

Ditinjau dari keterhubungan dalam konteks Twitter, suatu objek (*thing*) dapat memiliki atribut dan relasi. Atribut berisi pengenalan (*id*), nama (*name*), deskripsi (*description*) dan jenis (*type*). Sedangkan relasi berisi *metadata*, keadaan (*state*), mengikuti (*following*) dan pengikut (*follower*). Jika objek pertama mengambil nilai dari objek kedua, maka dikatakan bahwa objek pertama mengikuti (*following*) objek kedua. Ada pun objek kedua memiliki pengikut (*follower*) yaitu objek pertama. Relasi seperti inilah yang sebenarnya juga terdapat di dalam berbagai objek yang menyusun sistem *Internet of Things* (Almeida, dkk, 2013).

Aplikasi Twitter dalam Sistem Waktu-Nyata

Karakteristik dari Twitter yang memberikan informasi dalam waktu terkini membuat media sosial ini sesuai diterapkan dalam sistem waktu-nyata. Sebagai contohnya adalah dalam bidang manajemen inventaris. Dalam perusahaan penjualan eceran (*retail*), informasi tentang banyaknya stok, jumlah pesanan hingga tanggal kadaluarsa suatu barang harus terus-menerus diketahui nilainya. Pada kenyataannya, pelaporan inventaris tidak selalu diperbarui sesuai kondisi yang sebenarnya. Integrasi teknologi IoT ke dalam manajemen inventaris dapat memberikan jalan keluar terhadap permasalahan tersebut. RFID yang disematkan ke dalam tiap barang memberikan informasi terkini tentang keberadaan barang tersebut. Komunikasi antara barang yang bersematkan RFID dengan petugas dilakukan melalui *Extensible Messaging and Presence Protocol* (XMPP). Petugas sesuai dengan ruang lingkup pekerjaannya akan mendapatkan informasi terkini tentang kondisi inventaris saat itu melalui Twitter (Mathaba, dkk, 2012).

Aplikasi Twitter lainnya adalah berbentuk lingkungan perkantoran yang di dalamnya disematkan berbagai objek cerdas atau disebut juga *cognitive office*. Salah satu objek tersebut adalah sensor kelembaban yang diletakkan pada tanaman di dalam ruangan kantor. Sensor tersebut mengirimkan nilai hasil pengukuran kelembaban ke sebuah akun Twitter yang dimiliki oleh petugas ruangan.

Status Twitter kemudian juga dibaca melalui suatu skrip Python untuk mengaktifkan sebuah pengendali. Berdasarkan informasi dari Twitter tersebut pengendali menyalakan sumber cahaya ultraviolet (UV) hingga mencukupi kebutuhan dari tanaman (Kranz, dkk, 2010).

Pada kedua contoh tersebut, Twitter memiliki dua peranan. Pertama, yaitu sebagai media untuk menampilkan informasi terbaru yang dikirimkan dari hasil pengolahan suatu perangkat. Kedua, sebagai sumber informasi yang “dibaca” oleh perangkat untuk menjalankan suatu tindakan tertentu.

Motivasi

Twitter dapat dimanfaatkan untuk menjadi bagian pada sebuah sistem pengendalian jarak jauh melalui internet. Dengan beragamnya aplikasi klien untuk Twitter baik di PC maupun pada ponsel cerdas, pengguna sistem dapat mengendalikan perangkat secara mudah dan praktis. Tujuan dari penelitian ini adalah melakukan rancang bangun sebuah prototipe sistem pengaturan penyalan lampu menggunakan Twitter.

METODE

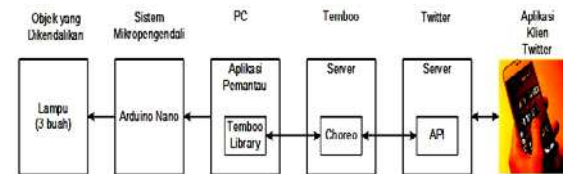
Rancang bangun untuk prototipe sistem secara keseluruhan adalah mengikuti tahapan-tahapan sebagaimana berikut:

Perancangan Sistem

Sistem yang dirancang adalah seperti dalam Gambar 1. Pada kesempatan awal, pengguna melalui aplikasi klien Twitter di ponsel cerdas miliknya mengirimkan perintah dengan format tertentu dalam bentuk status teks. Status tersebut kemudian ditampilkan oleh server dari Twitter. Aplikasi pemantau pada PC melalui pustaka (*library*) Temboo mengirimkan perintah ke modul Choreo di server Temboo untuk mengecek status terkini pada suatu akun Twitter. Choreo selanjutnya memanggil API dari Twitter untuk melakukan hal tersebut. Status terkini lalu dikirim balik ke server Temboo untuk selanjutnya diteruskan ke aplikasi pemantau. Seluruh proses yang terjadi antara PC, server Temboo dan server Twitter menggunakan protokol TCP/IP.

Ada pun sistem mikropengendali terus-menerus melakukan pengecekan melalui komunikasi serial (USART) pada salah satu port USB di PC jika terdapat kiriman status terbaru dari aplikasi pemantau. Setelah

diterima, mikropengendali selanjutnya memilah isi dari status tersebut dan mengatur penyalan lampu.

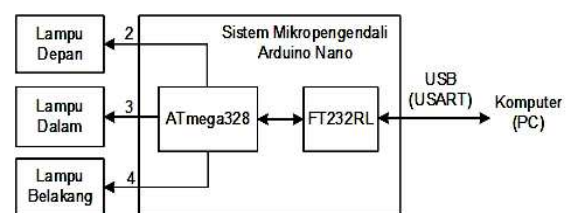


Gambar 1. Sistem Keseluruhan

Perancangan Perangkat Sistem Mikropengendali

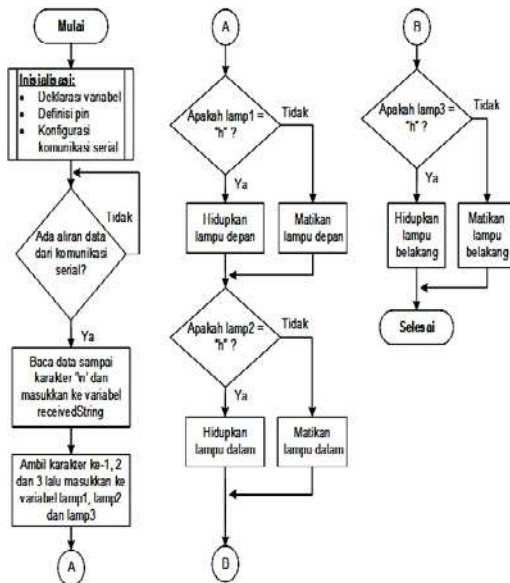
Sistem mikropengendali yang digunakan untuk penelitian ini adalah Arduino Nano. Di dalamnya terdapat mikropengendali ATMEL AVR ATmega328 sebagai pemroses utama dan cip FTDI FT232RL untuk menangani komunikasi serial terhadap PC. ATmega328 memiliki memori program (*flash*) sebesar 32 KB, EEPROM 1 KB dan SRAM 2 KB (ATMEL, 2014) yang dipandang sudah mencukupi untuk kebutuhan penelitian ini. Ada pun cip FT232RL memiliki dukungan penuh terhadap mode komunikasi Universal Synchronous/Asynchronous Receiver/Transmitter (USART) dengan *bit rate* dari 300 baud hingga 3 Mbaud dan didukung oleh sistem operasi Windows, Mac maupun Linux (Future Technology Devices, 2010).

Pada penelitian ini dicontohkan bahwa lampu rumah yang dikendalikan terdiri dari tiga buah, yaitu lampu depan, lampu dalam dan lampu belakang. Ketiga lampu tersebut diwakili oleh LED yang dihubungkan ke mikropengendali ATmega328 di bagian *Digital Input* pada pin 2, 3 dan 4. Ada pun cip FT232RL dihubungkan ke PC melalui salah satu port USB (yang terdeteksi di sistem operasi sebagai COM14). Komunikasi serial yang terjadi antara mikropengendali dan PC menggunakan mode USART dengan konfigurasi 8 bit data, 1 bit stop dan *bit rate* sebesar 9600 bps. Koneksi sistem mikropengendali dengan bagian-bagian lain adalah seperti diperlihatkan dalam Gambar 2.



Gambar 2. Koneksi Sistem Mikropengendali

Program untuk mikropengendali atau *firmware* ditulis menggunakan bahasa pemrograman Arduino melalui aplikasi Arduino IDE versi 1.6.5. Alur dari program tersebut seperti ditunjukkan dalam Gambar 3.



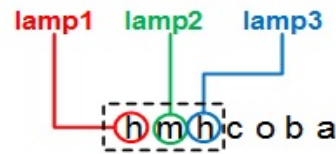
Gambar 3. Koneksi Sistem Mikropengendali

Pada tahap inisialisasi dilakukan proses-proses deklarasi variabel. Yaitu variabel lamp1 untuk lampu depan, lamp2 untuk lampu dalam dan lamp3 untuk lampu belakang serta *receivedString* untuk menampung status Twitter. Sesudah itu dilakukan pendefinisian pin sebagai keluaran (*output*) atau masukan (*input*). Dalam hal ini pin *Digital Input* nomor 2, 3 dan 4 dijadikan sebagai keluaran, yang masing-masing terhubung dengan LED untuk lampu depan, lampu dalam dan lampu belakang. Inisialisasi berakhir dengan pengaturan komunikasi serial untuk menentukan nilai *bit rate*.

Program berlanjut dengan melakukan pengecekan aliran data dalam komunikasi serial. Data tersebut berasal dari aplikasi pemantau pada PC yang mengirimkan isi status Twitter terbaru. Proses ini berulang terus manakala tidak dijumpai data dan baru berlanjut ke tahap berikutnya jika terdapat data.

Data berupa *string* yang diterima lalu dibaca hingga menemui karakter ganti baris (*newline*) atau '\n'. *String* tersebut lalu dimasukkan ke variabel *receivedString* untuk diteruskan dengan proses pemilahan (*parsing*). Dalam proses pemilahan, hanya tiga karakter terdepan yang akan diambil. Karakter pertama,

kedua dan ketiga, masing-masing akan dimasukkan ke variabel lamp1, lamp2 dan lamp3 seperti pada Gambar 4. Karakter 'h' menandakan bahwa lampu hidup sedangkan 'm' berarti lampu mati.



Gambar 4. Pemilahan Karakter

Berdasarkan isi dari ketiga variabel tersebut, maka mikropengendali memutuskan apakah akan mematikan atau menyalakan lampu. Sesuai contoh pada Gambar 4 di atas, maka mikropengendali akan menghidupkan lampu depan, mematikan lampu dalam dan menyalakan lampu belakang.

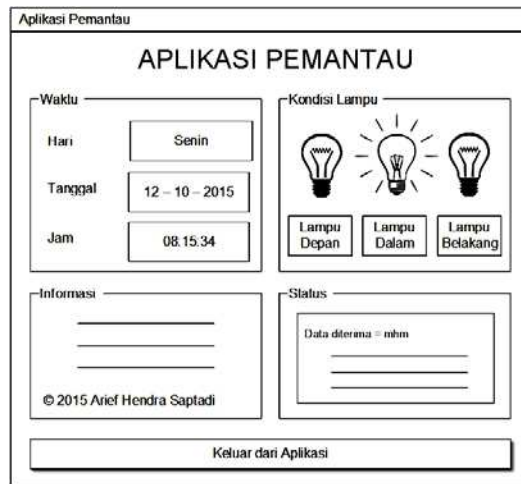
Pengembangan Aplikasi Pemantau pada PC

Aplikasi pemantau atau *monitoring application* pada PC sepenuhnya ditulis menggunakan bahasa pemrograman Processing (<http://processing.org>). Processing awalnya dikembangkan oleh Casey Reas dan Ben Frye di MIT Media Lab pada tahun 2001. Processing adalah bahasa pemrograman berkonteks grafis yang merupakan turunan dari Java. Lingkungan atau aplikasi pemrograman untuk Processing disebut *Processing Development Environment* atau disingkat PDE (Reas dan Frye, 2010). PDE yang digunakan untuk membuat aplikasi pemantau adalah versi 2.2.1.

Aplikasi pemantau mengakses Twitter API melalui layanan web bernama Temboo (<http://temboo.com>). Temboo menyediakan modul untuk mengakses API dari berbagai situs. Modul tersebut dinamakan Choreo. Melalui Choreo, pengembang aplikasi cukup memasukkan data yang diperlukan dan kode program kemudian dapat langsung dihasilkan. Kode ini kemudian disematkan ke dalam aplikasi.

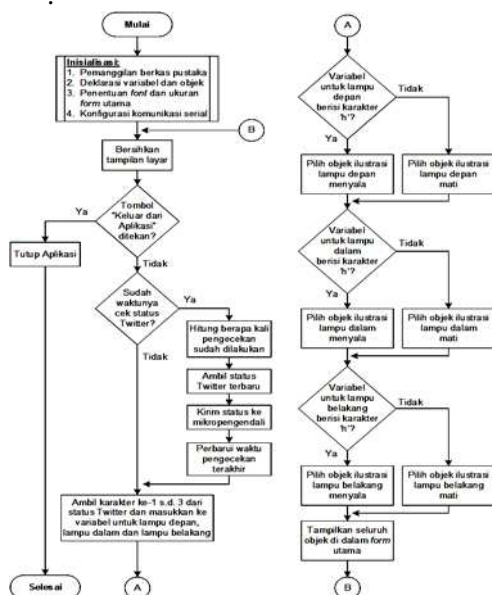
Rancangan tampilan untuk aplikasi pemantau adalah seperti pada Gambar 5. Seluruh objek di dalam *form* utama dibuat melalui berbagai rutin terpisah. Beberapa objek merupakan gabungan dari dua objek terpisah. Seperti halnya tombol perintah (*command button*) yang terdiri dari objek

persegi panjang (*rectangle*) dan label teks (*text label*) di atasnya.



Gambar 5. Rancangan Tampilan Aplikasi

Diagram alir untuk aplikasi secara keseluruhan adalah seperti diperlihatkan pada Gambar 6. Pada tahap inisialisasi terdapat empat proses yang berlangsung. Pertama, aplikasi memanggil seluruh berkas pustaka yang diperlukan untuk menampilkan tanggal dan waktu, komunikasi serial dan koneksi ke server Temboo. Kedua, aplikasi melakukan deklarasi untuk seluruh variabel dan objek, termasuk di dalamnya. Objek yang dimaksud adalah port serial, *font* dan gambar. Ketiga, aplikasi menentukan ukuran form utama dan mendefinisikan jenis *font* yang akan dipakai. Terakhir, aplikasi memilih port serial dan nilai *bit rate* yang digunakan



Gambar 6. Diagram Alir Aplikasi Pemantau

Setelah tahap inisialisasi selesai, berikutnya aplikasi membersihkan tampilan layar lalu melakukan pengecekan jika terdapat penekanan pada tombol “Keluar dari Aplikasi” (seperti dalam Gambar 5). Jika tombol ini ditekan, maka aplikasi akan ditutup.

Jika aplikasi masih dijalankan, maka berlangsung pengecekan berikutnya. Yaitu, apakah sudah tiba saatnya untuk mengecek status Twitter. Bila memang sudah waktunya maka akan dihitung berapa kali pengecekan sudah dilakukan. Selanjutnya aplikasi mengambil status Tweet terbaru. Proses ini melibatkan Choreo di Temboo untuk mengakses Twitter API. Data yang dimasukkan ke dalam Choreo tersebut adalah terdiri dari AccessToken, AccessTokenSecret, ConsumerKey, ConsumerSecret dan Query (Temboo, 2015). Empat data yang pertama diperoleh ketika mendaftarkan aplikasi ke dalam akun Twitter. Sedangkan Query yang dimasukkan adalah nama akun di Twitter tersebut yaitu IOTExperiments. Ada pun API yang diakses adalah berupa perintah GET search/tweets. Perintah ini merupakan bagian dari Twitter Search API dan berfungsi untuk memperoleh sekelompok *tweet* yang relevan dengan *query* pencarian (Twitter, 2015).

Sesudah didapatkan, teks status selanjutnya dikirimkan ke sistem mikropengendali melalui komunikasi serial. Kemudian catatan waktu pengecekan terakhir diperbarui. Proses berlanjut dengan mengambil tiga karakter terdepan dari *tweet* yang dimasukkan ke tiga variabel, masing-masing satu karakter untuk lampu depan, lampu dalam dan lampu belakang.

Pada tiap variabel tersebut dilakukan pengecekan apakah berisi nilai 'h'. Jika demikian, maka pada aplikasi akan memilih gambar berupa lampu menyala. Jika berisi karakter lain atau 'm', maka gambar berupa lampu mati akan dipilih. Proses ini dijalankan untuk ketiga lampu tadi. Terakhir, seluruh objek grafis di dalam *form* utama (termasuk gambar lampu menyala atau mati) ditampilkan. Alur program kemudian kembali lagi ke keadaan dimana tampilan layar dibersihkan. Demikian seterusnya hingga aplikasi ditutup.

HASIL DAN PEMBAHASAN

Dalam upaya untuk mendapatkan hasil akhir dari proses perancangan secara

keseluruhan maka dilakukan tiga jenis pengujian, yaitu pengujian pada sistem mikropengendali, aplikasi pemantau dan *query* pada server Temboo. Pengujian melibatkan perangkat *notebook* PC dengan spesifikasi prosesor Intel Core i5-3337U 1,8 GHz, RAM 4 GB dan HDD 750 GB. Sistem operasi yang digunakan adalah Windows 10 Pro 64 bit.

Sistem Mikropengendali

Pengujian ini bertujuan untuk mengetahui apakah sistem mikropengendali dapat menerima data dari PC dan mengatur penyalan lampu berdasarkan data tersebut. Aplikasi Arduino IDE versi 1.6.5 digunakan dengan memanfaatkan fitur Serial Monitor di dalamnya untuk memantau aliran data pada komunikasi serial.

Dalam pengujian ini sistem mikropengendali dihubungkan ke PC dan terdeteksi di port COM14. Pada Serial Monitor diketikkan 8 kombinasi perintah, satu demi satu yang masing-masing terdiri dari tiga karakter, untuk mengatur penyalan tiga buah lampu. Sistem mikropengendali dikatakan dapat menerima data dari PC, jika mikropengendali mengirimkan balasan berupa karakter yang diketik tersebut. Mikropengendali dianggap berhasil mengatur penyalan lampu bila ketiga lampu tersebut dinyalakan maupun dimatikan sesuai dengan data yang diperoleh.

Tampilan dari Serial Monitor dan respon pengaturan penyalan lampu dari mikropengendali adalah seperti diperlihatkan dalam Gambar 7. Dari tampilan tersebut dapat diketahui bahwa ketika diketikkan karakter “hhh”, mikropengendali mengirimkan kembali karakter yang sama, memperlihatkan status penyalan dari masing-masing lampu sekaligus menyalakan ketiga LED yang mewakilinya.



Gambar 7. Tampilan Serial Monitor dan Respon Mikropengendali

Respon selengkapnya dari mikropengendali terhadap 8 buah kombinasi masukan adalah seperti ditunjukkan dalam Tabel 1. Berdasarkan tabel tersebut maka dapat disimpulkan bahwa mikropengendali telah mampu menerima data dari PC sekaligus mengatur penyalan lampu sesuai data yang diterima.

Tabel 1. Pengujian Sistem Mikropengendali

No	Data dari PC	Respon yang Diharapkan			Respon Mikropengendali			
		L 1	L 2	L 3	Tampilan Data	L 1	L 2	L 3
1.	mm	x	x	x	mmm	x	x	x
2.	mmh	x	x	✓	mmh	x	x	✓
3.	mhm	x	✓	x	mhm	x	✓	x
4.	mhh	x	✓	✓	mhh	x	✓	✓
5.	hmm	✓	x	x	hmm	✓	x	x
6.	hmh	✓	x	✓	hmh	✓	x	✓
7.	hhm	✓	✓	x	hhm	✓	✓	x
8.	hhh	✓	✓	✓	hhh	✓	✓	✓

Keterangan:

1. L1 = lampu depan, L2 = lampu dalam, L3 = lampu belakang

2. x = mati, ✓ = menyala

Aplikasi Pemantau

Pengujian untuk aplikasi pemantau meliputi dua aspek yaitu tampilan dan fungsionalitas. Pengujian aspek tampilan menggunakan acuan rancangan seperti dalam Gambar 5. Sedangkan pengujian pada aspek fungsionalitas akan mendasarkan pada tiga fungsi yang harus dipenuhi dari aplikasi tersebut, yaitu mengecek status Twitter, menayangkan data dari Twitter dan mengirimkan data ke sistem mikropengendali.

Tampilan dari aplikasi pemantau yang dihasilkan adalah seperti pada Gambar 8. Terdapat enam bagian yang diamati dari tampilan tersebut yaitu *form* utama, waktu, informasi, kondisi lampu, status dan tombol untuk keluar dari aplikasi.



Gambar 8. Tampilan Aplikasi Pemantau

Ada pun indikator yang menunjukkan bahwa keenam bagian tersebut sudah berfungsi dengan baik berikut hasil ujinya adalah sebagaimana dicantumkan dalam Tabel 2. Dari tabel tersebut dapat diketahui bahwa seluruh bagian dari tampilan sudah berhasil diwujudkan sesuai rancangan awal.

Tabel 2. Hasil Pengujian Tampilan

No.	Bagian	Indikator	Hasil Uji
1.	Form utama	Ukuran dan warna sesuai rancangan	OK
2.	Waktu	Hari, Tanggal dan Jam senantiasa diperbarui sesuai waktu saat itu	OK
3.	Kondisi Lampu	Lampu menyala atau mati sesuai isi <i>tweet</i>	OK
4.	Informasi	Teks informasi muncul sesuai rancangan	OK
5.	Status	Status Twitter dapat ditampilkan	OK
6.	Tombol Perintah	Ketika diklik, keluar dari aplikasi	OK

Khusus untuk pengujian di bagian Kondisi Lampu dan Status, isi *tweet* tersebut dibandingkan antara yang diperoleh pada aplikasi pemantau dengan yang tampil di dalam akun Twitter @IoTExperiments (<http://twitter.com/iotexperiments>) seperti dalam Gambar 9.



Gambar 9. Akun @IoTExperiments

Pengujian dari sisi fungsionalitas memberikan hasil seperti tercantum dalam Tabel 3. Sesuai hasil uji pada tabel tersebut maka dapat disimpulkan bahwa seluruh fungsi pada aplikasi sudah dapat berjalan dengan baik.

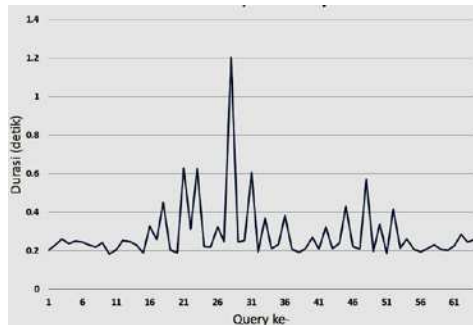
Tabel 3. Hasil Pengujian Fungsionalitas

No.	Fungsionalitas	Indikator	Hasil Uji
1.	Mengecek Status Twitter	Terdapat catatan di server Temboo bahwa ada aktivitas pengecekan status Twitter	OK
2.	Menayangkan Data dari Twitter	Tampilan Kondisi Lampu dan Status berubah sesuai isi <i>tweet</i>	OK
3.	Mengirim data ke Mikropengendali	Tiga lampu yang terhubung ke mikropengendali menyala atau mati sesuai isi <i>tweet</i>	OK

Query pada Server Temboo

Proses yang menentukan berhasil atau tidaknya pengaturan penyalan lampu terletak pada pengiriman *query* dari server Temboo menuju Twitter API. Setiap proses yang melibatkan interaksi antara Choreo dengan API dicatat oleh server Temboo ke dalam sebuah berkas log. Berdasarkan catatan pada log tersebut dapat ditentukan apakah *query* yang dikirimkan telah dijalankan dengan baik atau tidak sehingga menentukan nilai dari *success rate*. Di samping itu terdapat pula data yang menjelaskan seberapa lama *query* dari Temboo direspon oleh Twitter API. Data ini menentukan nilai dari durasi respon *query* dalam satuan detik.

Dalam pengujian ini, dilakukan pengiriman 64 buah *query* secara terus-menerus dalam jangka waktu kurang lebih satu jam. Di sisi lain, status dari akun @IoTExperiments juga dilakukan pengubahan sebanyak jumlah *query* tersebut. Berdasarkan catatan durasi respon pada server Temboo maka dapat dibuat grafik sebagaimana dalam Gambar 10 berikut ini.

Gambar 10. Durasi Respon *Query*

Data yang diperoleh menunjukkan bahwa durasi respon terlama adalah 1,204 detik sementara yang tercepat adalah 0,184 detik. Secara keseluruhan, nilai rata-rata durasi respon adalah sebesar 0,286 detik.

SIMPULAN DAN SARAN

Berdasarkan perancangan dan pengujian yang telah dilakukan maka dapat ditarik beberapa simpulan sebagaimana berikut ini. Pertama, proses komunikasi serial yang terjadi antara sistem mikropengendali dengan aplikasi pemantau telah berjalan dengan baik. Hal ini terbukti dari tampilan data dan respon pengaturan penyalan lampu yang sesuai dengan masukan yang diberikan dari PC. Kedua, seluruh bagian tampilan dari aplikasi pemantau sudah sesuai dengan rancangan awal. Hasil uji tampilan bersesuaian dengan indikator penilaian. Ketiga, dari sisi fungsionalitas, seluruh fungsi pada aplikasi pemantau telah terpenuhi. Dengan demikian aplikasi ini telah mendukung sepenuhnya pengendalian jarak jauh melalui status Twitter. Keempat, pengiriman query dari server Temboo menuju Twitter API memiliki tingkat keberhasilan sebesar 100% dengan rata-rata durasi respon sebesar 0,286 detik.

Pada pengembangan mendatang diharapkan sistem pengaturan penyalan lampu melalui status Twitter ini dapat mendukung penjadwalan (*scheduling*). Dengan fitur ini pemilik rumah yang akan meninggalkan rumah dalam waktu lama dapat mengatur penyalan lampu secara otomatis dengan pola-pola tertentu. Hal ini dimungkinkan mengingat bahwa Twitter mendukung teks status sepanjang 140 karakter yang semestinya mencukupi untuk keperluan tersebut.

DAFTAR PUSTAKA

- Almeida, R. A. P., Blackstock, M., Lea, R., Calderon, R., Prado, A. F. dan Guardia, H. C. 2013. Thing Broker: A Twitter for Things. Makalah yang disajikan dalam 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013), Zurich, Swiss, 8 – 12 September.
- ATMEL. 2014. *ATmega48A/PA/88A/PA/168A/PA328/P. ATMEL 8-bit Microcontroller with 4/8/16/32Kbytes In-System Programmable Flash. Datasheet Summary*. San Jose, California, USA: ATMEL Corporation.
- Future Technology Devices. 2010. *FT232R USB UART IC*. Future Technology Devices International Limited.
- Gubbi, J., Buyya, R., Marusic, S. dan Palaniswami, M. 2013. Internet of Things (IoT): A Vision, Architectural Elements and Future Directions. *Future Generation Computer Systems*, Volume 29, Issue 7, September 2013: 1645 – 1660.
- Inmobi dan Decisionfuel. *Global Mobile Media Consumption : A 'New Wave' Takes Shape*. Wave 3. Februari 2014.
- Kranz, M., Roalter, L. dan Michahelles, F. 2010. Things That Twitter: Social Networks and the Internet of Things. Makalah yang disajikan dalam What can the Internet of Things do for The Citizen (CioT) Workshop pada The Eighth International Conference on Pervasive Computing (Pervasive 2010), Helsinki, Finlandia, 17 – 20 Mei.
- Mathaba, S., Dlodlo, N., Smith, A., Makitla, I., Sibiya, G. dan Adigun, M. 2012. Interfacing Internet of Things Technologies of RFID, XMPP and Twitter to Reduce Inaccuracies in Inventory Management. Makalah yang disajikan dalam IST Africa 2012, Dar es Salaam, Tanzania, 9 - 11 Mei.
- Reas, C. dan Frye, B. 2010. *Getting Started with Processing*. Sebastopol, California, USA: O'Reilly Media, Inc.
- Statista. *Number of Internet Users in Indonesia from 2013 to 2018 (in millions)*. Didapat dari: <http://www.statista.com/statistics/254456/number-of-internet-users-in-indonesia/> pada 13 Oktober 2015.

Temboo. *Twitter.Search.LatestTweet*. Didapat dari: <https://temboo.com/library/Library/Twitter/Search/LatestTweet/> pada 14 September 2015.

Twitter. *GET search/tweets*. Didapat dari: <https://dev.twitter.com/rest/reference/get/search/tweets> pada 14 September 2015.

Yonezawa, T. dan Tokuda, H. 2010. *Twitthings: Sharing, Discovering and Defining Things' Happening Using Wireless Sensor Networks*. Makalah disajikan dalam Internet of Things 2010 International Conference, Tokyo, 29 November – 1 Desember.